(12) UK Patent Application (19) GB (11) 2 330 972 (13) A

(43) Date of A Publication 05.05.1999

(21) Application No 9811620.5

(22) Date of Filing 29.05.1998

(30) Priority Data
(31) 08959604 (32) 28.10.1997 (33) US

(71) Applicant(s)
Hewlett-Packard Company
(Incorporated in USA - California)
3000 Hanover Street, Palo Alto, CA 94304,
United States of America

(72) Inventor(s)
Robert G Gann
Steven L Webb
Greg A Degi
Gregory A Brake
Kevin J Youngers
Dan Scott Johnson

(74) Agent and/or Address for Service
Williams, Powell & Associates
4 St Paul's Churchyard, LONDON, EC4M 8AY,
United Kingdom

(51) INT CL^6
H04N 1/48 , G06K 9/20 , H04N 1/64

(52) UK CL (Edition Q)
H4F FFF FS1 FS25R FS30B FS30D FS30E FS30J FS30K
FS30P FS30X FS53D FS83B
G4R RET

(56) Documents Cited
GB 2321361 A GB 2143702 A

(58) Field of Search
UK CL (Edition P) G4R RET REX , H4F FCA FCCB FCCX
FCCY FFF FFS FFX
INT CL^6 G06K 9/00 9/20 9/60 9/78 , H04N 1/00 1/04
1/40 1/41 1/46 1/48 1/64
Online:WPI

(54) Abstract Title
Multiple image scanner

(57) A multiple image scanner performs a single scan of a document containing different types of images (e.g., text and graphics) and sends multiple renditions (132, 134) of the same document from the scanner to a host computer (102) (one high resolution low bit depth image (134), and one low resolution high bit depth image (132)), thus greatly reducing the total amount of data sent to and processed by the host computer (102).

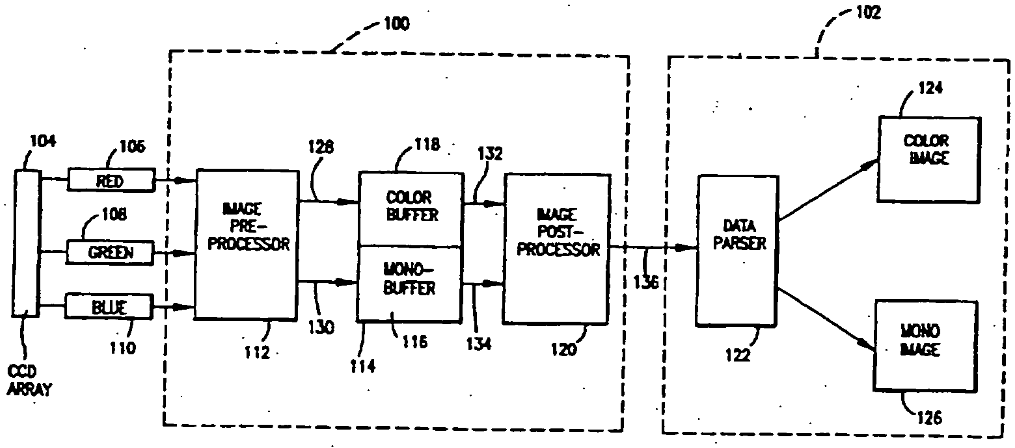

FIG.1

The specification as filed includes a computer program which is not reproduced here; it may be inspected in accordance with Section 118 of the Patents Act 1977.

This print takes account of replacement documents submitted after the date of filing to enable the application to comply with the formal requirements of the Patents Rules 1995

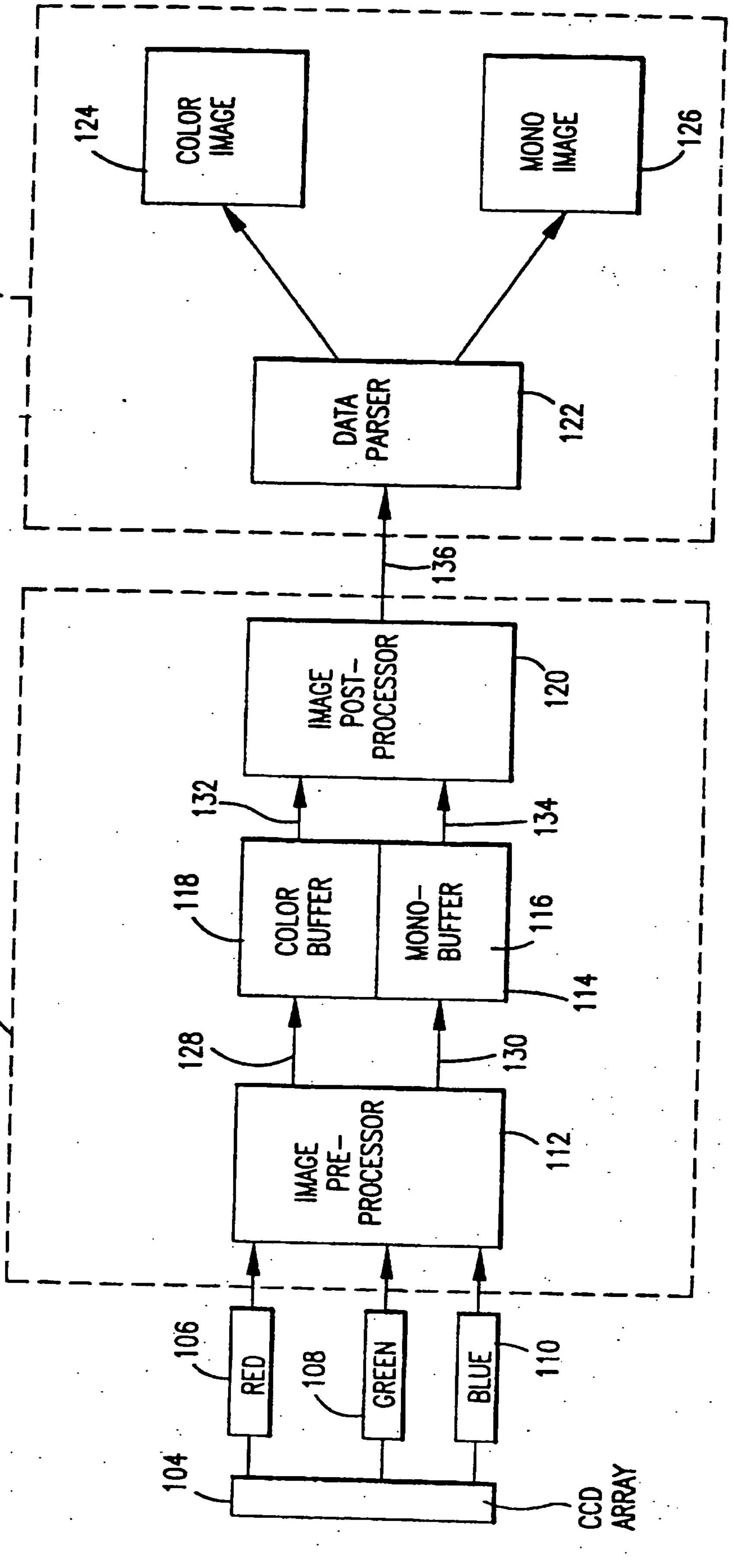
104
CCD ARRAY
106
RED
108
GREEN
110
BLUE
100
IMAGE PRE-PROCESSOR
112
128
130
COLOR BUFFER
118
MONO-BUFFER
116
114
132
134
IMAGE POST-PROCESSOR
120
136
102
DATA PARSER
122
COLOR IMAGE
124
MONO IMAGE
126

FIG.1

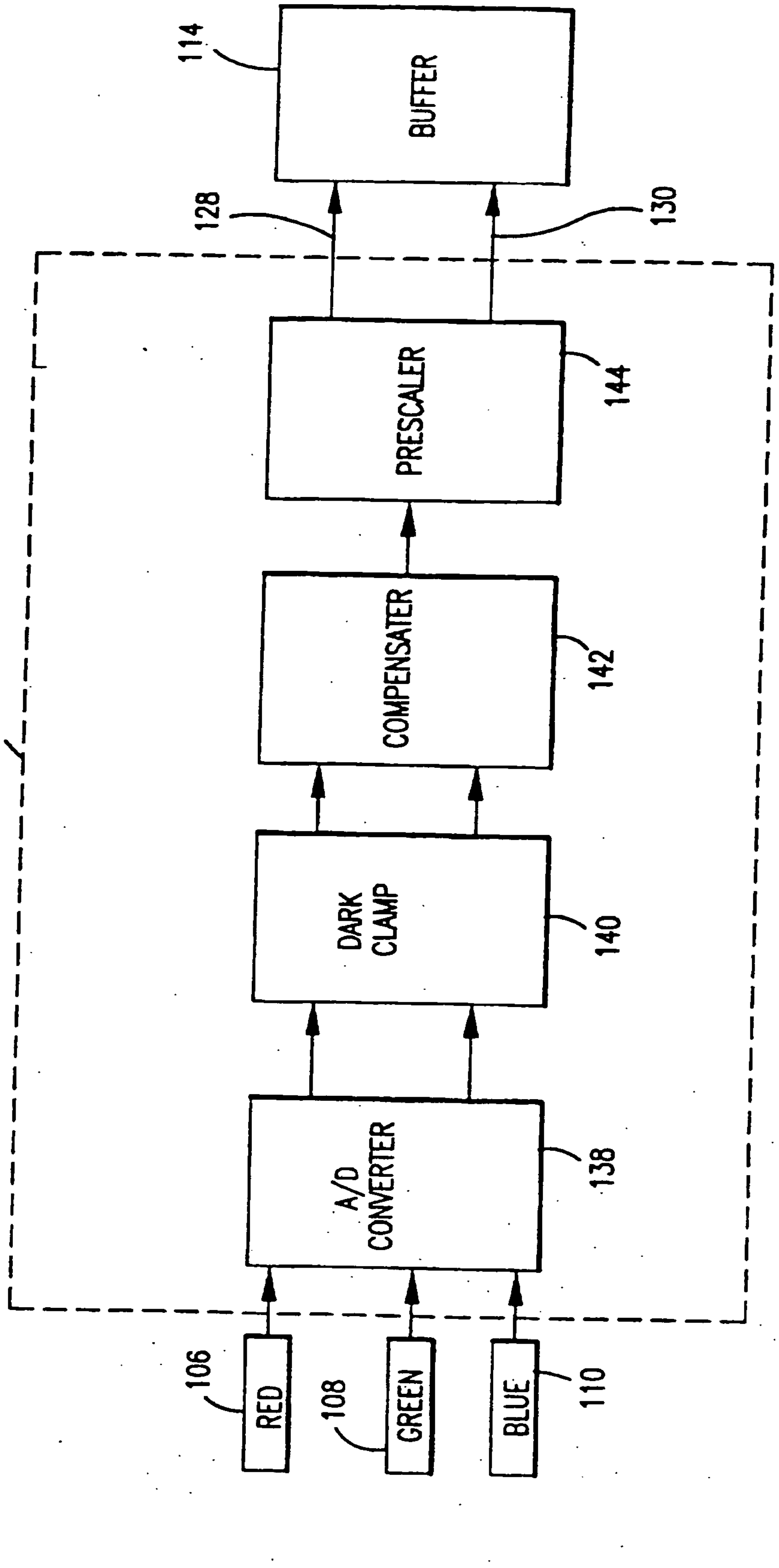
114
BUFFER
128
130
PRESCALER
144
COMPENSATER
142
112
DARK CLAMP
140
A/D CONVERTER
138
106
RED
108
GREEN
BLUE
110

FIG.2

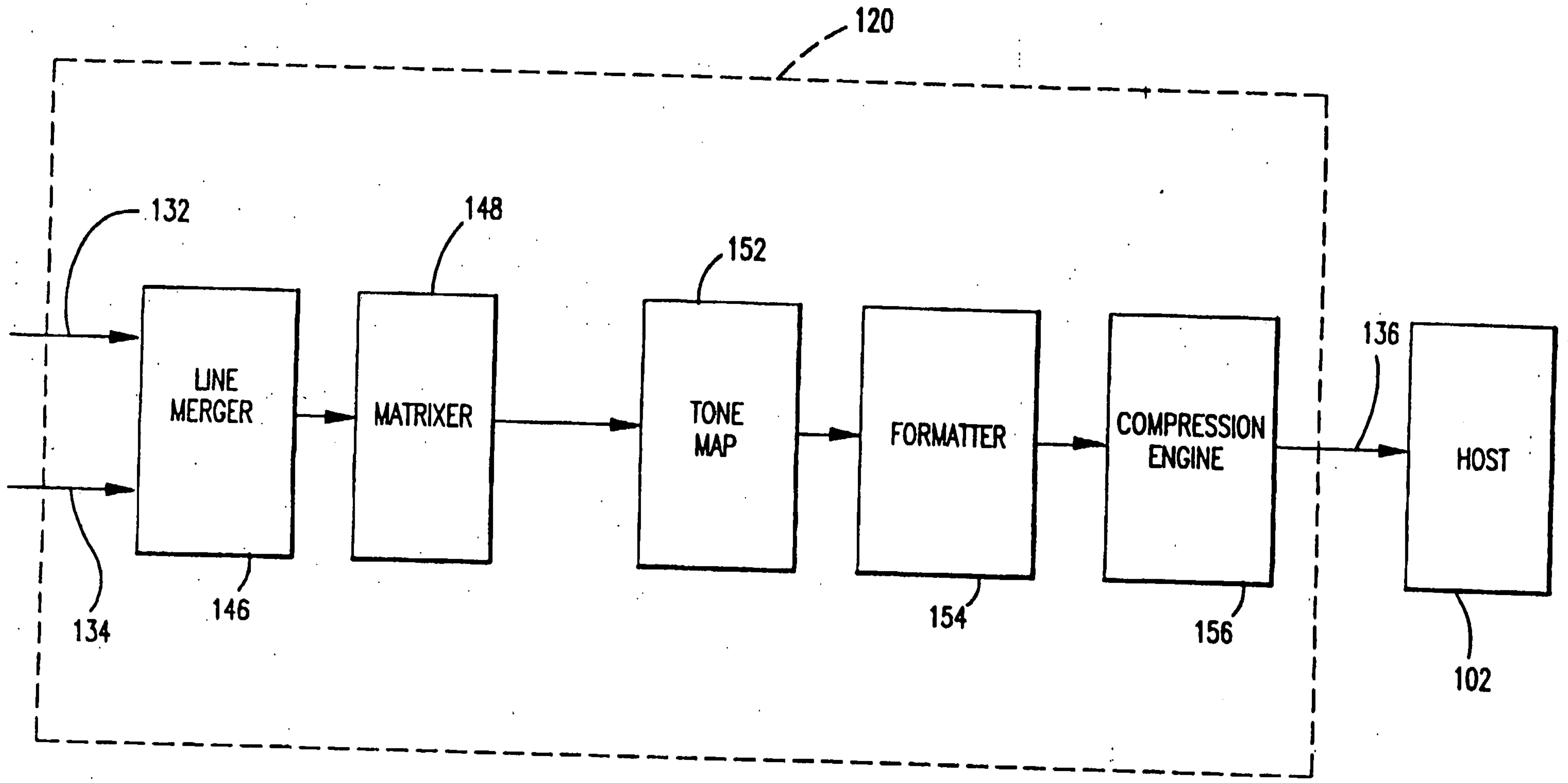
120
132
148
152
136
LINE MERGER
MATRIXER
TONE MAP
FORMATTER
COMPRESSION ENGINE
HOST
134
146
154
156
102

FIG.3

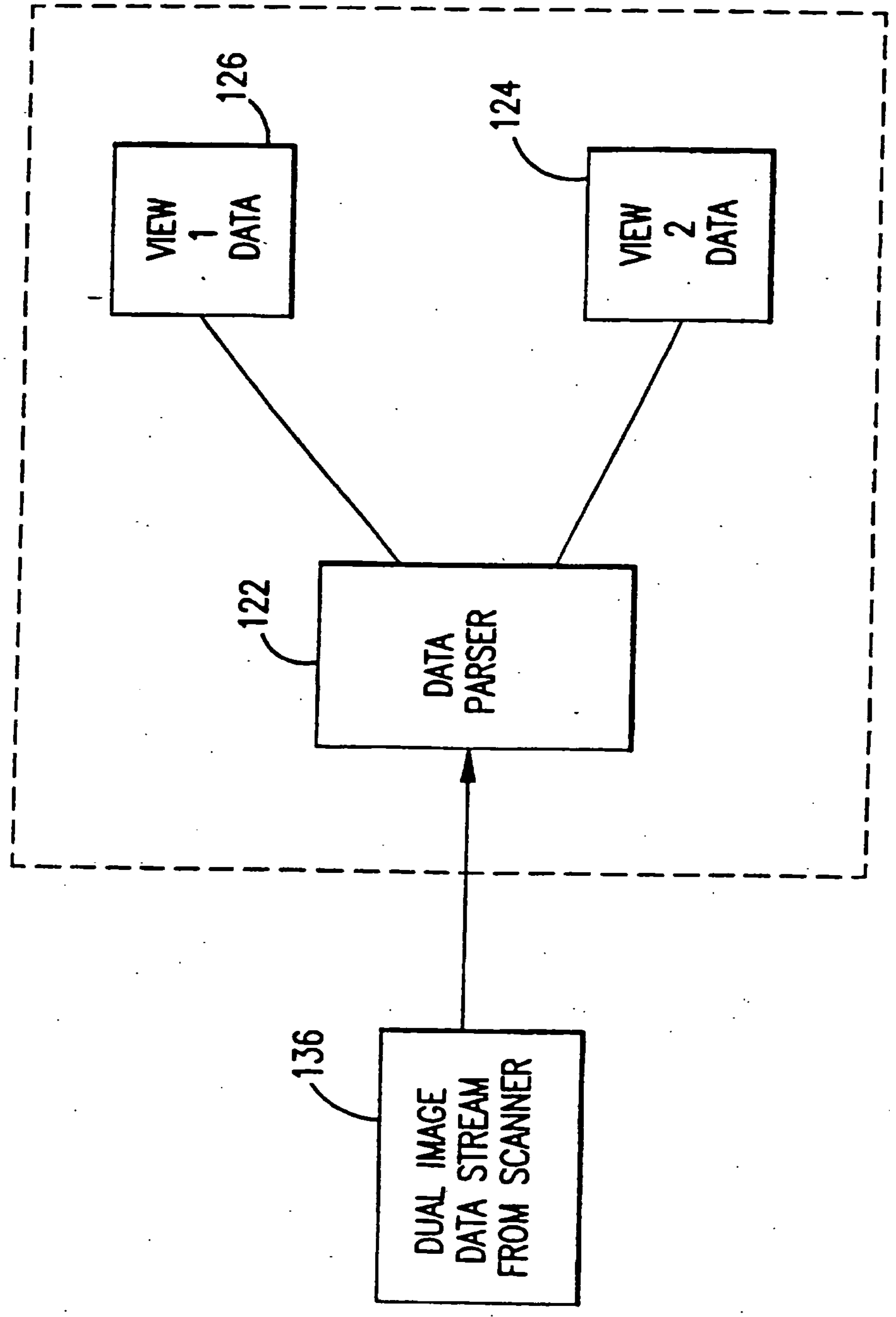
DUAL IMAGE DATA STREAM FROM SCANNER
136
122
DATA PARSER
126
VIEW 1 DATA
124
VIEW 2 DATA

FIG.4

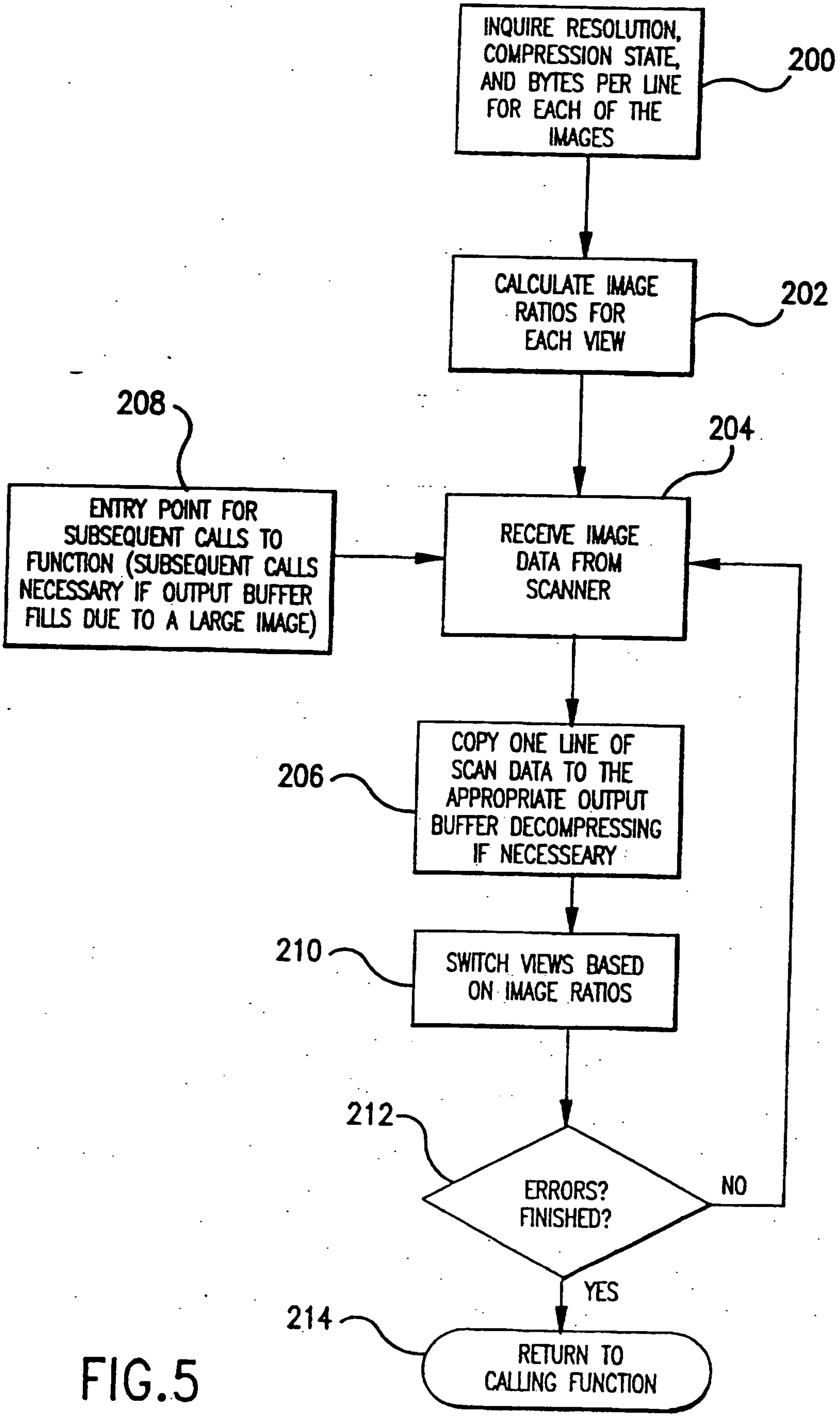
INQUIRE RESOLUTION, COMPRESSION STATE, AND BYTES PER LINE FOR EACH OF THE IMAGES
200
CALCULATE IMAGE RATIOS FOR EACH VIEW
202
208
ENTRY POINT FOR SUBSEQUENT CALLS TO FUNCTION (SUBSEQUENT CALLS NECESSARY IF OUTPUT BUFFER FILLS DUE TO A LARGE IMAGE)
204
RECEIVE IMAGE DATA FROM SCANNER
206
COPY ONE LINE OF SCAN DATA TO THE APPROPRIATE OUTPUT BUFFER DECOMPRESSING IF NECESSEARY
210
SWITCH VIEWS BASED ON IMAGE RATIOS
212
ERRORS? FINISHED?
NO
YES
214
RETURN TO CALLING FUNCTION

FIG.5

MULTIPLE IMAGE SCANNER 2330972

The present invention relates generally to the field of optical scanners and more particularly to a method and system for obtaining multiple images from a single scan. In particular, this invention provides a means for obtaining color image data and black and white image data from a single scan.

Optical scanners are used to capture and digitize images. For example, an optical scanner can be used to capture the image of printed matter on a sheet of paper. The digitized image can then be electronically stored and/or processed with character recognition software to produce ASCII text. Most optical scanners use illumination and optical systems to illuminate the object and focus a small area of the illuminated object, usually referred to as a "scan line," onto the optical photosensor array. The entire object is then scanned by sweeping the illuminated scan line across the entire object, either by moving the object with respect to the illumination and optical assemblies or by moving the illumination and optical assemblies relative to the object.

A typical scanner optical system will include a lens assembly to focus the image of the illuminated scan line onto the surface of the optical photosensor array. Depending on the particular design, the scanner optical system may also include a plurality of mirrors to "fold" the path of the light beam, thus allowing the optical system to be conveniently mounted within a relatively small enclosure.

While various types of photosensor devices may be used in optical scanners, a commonly used sensor is the charge coupled device or CCD. As is well-known, a CCD may comprise a large number of individual cells or "pixels," each of which collects or builds-up an electrical charge in response to exposure to light. Since the size of the accumulated electrical charge in any given cell or pixel is related to the intensity and duration of the light exposure, a CCD may be used to detect light and dark spots on an image focused thereon. In a typical scanner application, the charge built up in each of the CCD cells or pixels is measured and then discharged at regular intervals known as exposure times or sampling intervals, which may be about 5 milliseconds or so for a typical scanner. Since the charges (i.e., image data) are simultaneously collected in the CCD cells during the exposure time, the CCD also includes an analog shift register to convert the simultaneous or parallel data from the CCD cells into a sequential or serial data stream.

A typical analog shift register comprises a plurality of "charge transfer buckets" each of which is connected to an individual cell. At the end of the exposure time, the charges collected by each of the CCD cells are simultaneously transferred to the charge transfer buckets, thus preparing the CCD cells for the next exposure sequence. The charge in each bucket is then transferred from bucket to bucket out of the shift register in a sequential or "bucket brigade" fashion during the time the CCD cells are being exposed to the next scan line. The sequentially arranged charges from the CCD cells may then be converted, one-by-one, into a digital signal by a suitable analog-to-digital converter.

In most optical scanner applications, each of the individual pixels in the CCD are arranged end-to-end, thus forming a linear array. Each pixel in the CCD array thus corresponds to a related pixel portion of the illuminated scan line. The individual pixels in the linear photosensor array are generally aligned in the "cross" direction, i.e., perpendicular to the direction of movement of the illuminated scan line across the object (also known as the "scan direction"). Each pixel of the linear photosensor array thus has a length measured in the cross direction and a width measured in the scan direction. In most CCD arrays the length and width of the pixels are equal, typically being about 8 microns or so in each dimension.

The sampling rate in the cross direction is a function of the number of individual cells in the CCD. For example, a commonly used CCD photosensor array contains a sufficient number of individual cells or pixels to allow a sampling rate in the cross direction of about 600 pixels, or dots, per inch (600 ppi), which is referred to herein as the native sampling rate in the cross direction.

The sampling rate in the scan direction is inversely related to the product of the scan line sweep rate and the CCD exposure time (i.e., the sampling interval). Therefore, the sampling rate in the scan direction may be increased by decreasing the scan line sweep rate, the CCD exposure time, or both. Conversely, the sampling rate in the scan direction may be decreased by increasing the scan line sweep rate, the CCD exposure time, or both. The "minimum sampling rate in the scan direction" for a given exposure time is that sampling rate achieved when scanning at the maximum scan line sweep rate at that exposure time. For example, a maximum scan line sweep rate of about 3.33 inches per second and a maximum exposure time of about 5 milliseconds will result in a minimum sampling rate in the scan direction of about 60 ppi.

Currently, optical character recognition (OCR) requires 300 ppi sampling rates for accurate results. Thus, a 300 ppi 4 bit gray scan (8.5 X 11), which is high resolution, low bit depth, is approximately 4.2 Megabytes. Color fidelity requires a 24 bit color scan. Thus, a 150 ppi 24 bit color scan (8.5 X 11), which is low resolution, high bit depth, is approximately 6.3 Megabytes. In order to provide a scan of a document that has both color pictures or drawings and writing requiring OCR, the scan would have to be approximately 300 ppi at 24 bits (8.5 X 11) which corresponds to 25.24 Megabytes of memory. Accordingly, to scan a document that includes both text and pictures would require quite a bit of memory. Yet, the software on the computer will down sample the color image to approximately 6.3 Megabytes and throw away the color image to obtain the text. This process is extremely slow to perform in software and unnecessarily consumes a great deal of memory. Another alternative is to first scan either the text or the graphics and then perform a scan of the other. Then the document could be regenerated by software. However, this is also a very time consuming method of scanning the document, besides using a lot of memory as well.

None of the above systems provide a scanner that is able to scan a document containing both text and graphics, and greatly reduce the total amount of data being sent from the scanner to the host computer (which is currently a speed constraint), and reduce the total amount of data being stored and processed by the host computer software.

The present invention seeks to provide improved image scanners.

According to an aspect of the present invention there is provided a method for generating multiple renditions of an image with a scanner, said method comprising the following steps:

(a) performing a single scan of said image with said scanner; and

(b) said scanner generating more than one rendition of said image obtained from said single scan of said image, said more than one rendition comprising at least one high resolution, low bit depth gray scale image and at least one low resolution, high bit depth gray scale image.

Other aspects of the present invention will be evident from claims 2 and 3.

Preferred embodiments provide a multiple image scanner that performs a single scan of a document containing multiple types of images (e.g., text and graphics) and send multiple renditions of the same document from the scanner to the host computer (e.g., one high resolution gray scale image, and one low resolution high bit depth colour image), thus greatly reducing the total amount of data sent to and processed by the host computer, for example, from approximately 25.24 Megabytes to approximately 10.5 Megabytes (a 300 ppi 4 bit gray image [8.5 x 11] that is 4.2 Megabytes and a 150 ppi 24 bit colour [8.5 x 11] that is 6.3 Megabytes), which is a 2.4 to 1 reduction.

Preferred scanners send the multiple images to the host computer interleaved on a line basis, for example, with two lines of high resolution gray data and then one line of low resolution colour data, if the resolution ratio between the images was two. The advantage being that there is less information for the host software to process and store, and less information to be sent from the scanner to the host, which is a current limitation on scanner speed.

Preferred embodiments further comprise host computer software that is capable of parsing the data stream of interleaved images into the individual images that are then ready for manipulation and further processing by the host computer.

For generating multiple renditions of an image with a scanner, preferred methods comprise the following steps: performing a single scan of said image with said scanner; and said scanner generating more than one rendition of said image obtained from said single scan of said image. The step of generating more than one rendition of said image obtained from a single scan of said image may further comprise the step of generating at least one gray scale image and at least one color image, which may further comprise the step of generating at least one high resolution gray scale image and at least one low resolution, high bit color image. Once said scanner generates more than one rendition of said single scan image, said scanner may send said more than one rendition of said image obtained from a single scan of said image to a host computer for further processing.

Preferred scanners comprise an image processing unit for generating more than one rendition of an image from a single scan, wherein said image processing unit may comprise a prescaler, wherein said prescaler may split a single scan image into more than one rendition of said single scan image. Said prescaler may also split said single scan image into at least one monochrome rendition of said single scan image and at least one color rendition of said single scan image. Further, said prescaler may split said single scan image into at least one high resolution gray scale image and one low resolution, high bit color image. The prescaler may also split said single scan image into at least one high resolution, low bit depth gray scale or monochrome image and at least one low resolution, high bit depth gray scale or monochrome image.

Exemplary embodiments of the present invention will now be described, by way of example only, with reference to the accompanying drawings in which:

Figure 1 shows a block diagram of the data path for a preferred multiple image scanner;

Figure 2 shows a more detailed block diagram of the data path of the image pre-processor for a preferred multiple image scanner;

Figure 3 shows a more detailed block diagram of the data path of the image post-processor for a preferred multiple image scanner;

Figure 4 shows a more detailed block diagram of the data path from the multiple image scanner to the host computer according to a preferred embodiment of the present invention; and

Figure 5 shows a flow chart of a method of parsing a data stream containing more than one set of data.

Preferred embodiments provide a scanner that is capable of obtaining two sets of scan data simultaneously from a single scan. While the region of the scan is the same, the data types and resolutions of the two views can vary. For example, when a scan of a document is performed, an image processing unit in the scanner will generate one high resolution gray scale image and one low resolution 24 bit color image of the document scanned and then send both renditions of the scanned document to a host computer interleaved in a data stream, which the host computer will then parse in order to recreate the two views of the scanned document.

Figure 1 shows a block diagram of the data path for a dual image scanner with dashed line area 100 being scanner functions implemented by an image processeing unit and dashed line area 102 being host computer functions. It should be noted that although the embodiment being described herein is a dual image scanner comprising a black and white and a color image, the concepts are extensible to multiple images. Image processing unit 100 is implemented in an ASIC with functional details described below with respect to Figures 2 and 3. When a document is scanned, a CCD array 104 or other known photosensor device outputs a red data signal 106, a blue data signal 108 and a green data signal 110. These three data signals are then converted into a color image data signal 128 and a monochrome image data signal 130 by an image pre-processor 112. The image pre-processor 112 places the color image data signal 128 into a color buffer portion 118 of buffer 114 and the monochrome image data signal 130 into a monochrome buffer portion 116 of buffer 114.

Next the image post-processor 120 uses the image data in the color buffer 118 and the monochrome buffer 116 to generate either 1 or 2 views of the window scanned by the scanner as specified by the host computer. If two views are generated, the image post-processor will send the two views to the host computer in an interleaved data stream 136.

When the host computer receives the two views of the scan window, a data parser 122 will parse the data stream 136 into a color image 124 and a monochrome image 126. The two views are then ready to be manipulated by the host computer. The advantage is less information for the host software to process and store, and less information to be sent from the scanner to the host, which is a current limitation on the speed of scanners.

I. MULTIPLE IMAGE SCANNER

The dual image scanner reduces the data required to be sent from the scanner to the host computer for complete page (text and graphics/image) information with one scan. Currently optical character recognition requires 300 ppi sampling rates for accurate results. For color fidelity, a 24 bit color scan is required. In order to do this, the typical scan would be 300 ppi at 24 bits (8.5 X 11), which is 25.24 Megabytes. By letting the scanner send two renditions of the same page, one high resolution gray scale image, and one low resolution 24 bit color image, the total amount of data sent to and processed by the host software is greatly reduced. A 300 ppi 4 bit gray (8.5 X 11) is approximately 4.2 Megabytes and a 150 ppi 24 bit color (8.5 X 11) is approximately 6.3 Megabytes for a total of 10.5 Megabytes, which is approximately a 2.4 to 1 reduction in data.

The preferred multiple image scanner has an image processing unit which performs the above operations very fast. The image processing unit may be implemented in a conventional field programmable gate array, an ASIC or the like, using verilog, for example with the function description provided herein. The image processing unit of the multiple image scanner comprises the image pre-processor 112, buffer 116 and image post-processor

120, which are described in more detail below.

A. Image Pre-Processor

Referring to Figure 2, the image pre-processor 112 takes the RGB (red 106, green 108 and blue 110) image data from the CCD 104 and places either one or two different representations of that image into the buffer 114. One of the possible representations is a color image and the other is a monochrome image which is generated by selecting only one of the three color channels (red, green, or blue). Other possibilities would be generating two different gray scale or monochrome images or generating two color images. The two different buffer images have the same y resolution but can have different x resolutions and bit depths. The bit depth of each image can be selected to be either 12-bit or 8-bit and the x resolution can be the CCD resolution divided by any integer factor from 1 to 63. The CCD resolution is the same for both images and can be either 300 ppi or 150 ppi. For example, the image pre-processor 112 can be programmed to put 300 ppi (x) by 300 ppi (y) 8-bit monochrome (green channel) data and 100 ppi (x) by 300 ppi (y) 12-bit color data into the buffer. Note that both images do not need to be generated. The pre-processor can be programmed to generate only the color representation or only the monochrome representation or both.

The four main functions performed by the image pre-processor of the present invention are described in greater detail below and with reference to Figures 1 and 2.

1. A/D Converter

The A/D convertor takes the 300 ppi or 150 ppi analog RGB image data from the CCD array and converts either 1 of the 3 or all 3 color values for each pixel to 10-bit digital numbers. These digital values are padded to 12 bits and sent to the dark clamp.

2. Dark Clamp

The dark clamp subtracts the average dark current for the line from each pixel value. Pixels inside the desired scan window are then sent to the compensator.

3. Compensator

The compensator removes pixel variations caused by non-uniform illumination and CCD response. Compensated pixel values are then sent to the prescaler.

4. Prescaler

The prescaler splits the compensated pixel values into a monochrome image and a color image, reduces the x-resolution of each image by a different integer factor, and puts one or both of the resulting images into the buffer. The prescaler splits the image data from the compensator into a monochrome stream and a color stream, reduces the x-resolution of each stream by a different integer factor, and then stores one or both of the resulting streams in the buffer. This is done to minimize the amount of buffer space consumed by each line of image data, especially when 2 views are requested by the host. For example, if no prescaler were provided and the host requested a 300 ppi monochrome view and a 100 ppi color view of an 8.5 inch wide scan window, each line of the image would consume 11.475 Kbytes of buffer space (8.5 inches 300 ppi 3 colors/pixel 1.5 bytes/color). However, with the prescaler each line consumes only 6.375 Kbytes (8.5 inches 300 ppi 1 color/pixel + 8.5 inches 100 ppi 3 colors/pixel 1.5 bytes/color).

Note that the prescaler can either place full 12-bit or truncated 8-bit data into the buffer. The data width can be specified independently for each stream. The prescaler reduces the x-resolution of the 2 data streams by averaging adjacent pixels together rather than simply dropping pixels. This effectively lowpass filters the image data before it is resampled at the lower resolution and helps reduce aliasing. The prescaler can reduce x-resolution by any integer factor between 1 and 63.

B. Image Post-Processor

The image post-processor uses the image(s) in the buffer to generate either 1 or 2 views of the scan window specified by the host. Each view is generated from either the color image or the monochrome image. The two views can be generated from the same or different images. For example, one view could be generated from the color image and the other view from the monochrome image OR both views could be generated from the color image OR both views could be generated from the monochrome image. The two views can have different y resolution but the y resolution of each view must be the buffer y resolution divided by an integer factor from 1 to 16. The two views can also have different x resolutions which can be any resolution form 1/4 to 2X the x resolution of the buffer image from which the view is generated. Each view can also have different matrices, tonemaps, data types, etc. (with some restrictions which will be covered in greater detail below).

When two views are enabled, the views will be sent out interleaved on a line basis dependent on the ratio of their y resolutions. For example, if view one's y resolution is 300 and view two's y resolution is 150, then the data would alternate with two lines of view one's image data and then one line of view two's image data (except at the beginning). Other examples of interleaved data as a ratio of y resolution:

(2 to 1 ratio) → output 300 to 150 ratio: 300, 150, 300, 300, 150, 300, 300, 150....
(1 to 2 ratio) → output 150 to 300 ratio: 150, 300, 300, 150, 300, 300, 150, 300...
(3 to 1 ratio) → output 300 to 100 ratio: 300, 300, 100, 300, 300, 300, 100, 300, 300, 300...
(1 to 3 ratio) → output 100 to 300 ratio: 300, 100, 300, 300, 300, 100, 300, 300, 300, 100...
(4 to 1 ratio) → output 300 to 75 ratio: 300, 300, 75, 300, 300, 300, 300, 75, 300, 300...
(1 to 4 ratio) → output 75 to 300 ratio: 300, 75, 300, 300, 300, 300, 75, 300, 300, 300...
(5 to 1 ratio) → output 300 to 60 ratio: 300, 300, 300, 60, 300, 300, 300, 300, 300, 60...
(1 to 5 ratio) → output 60 to 300 ratio: 300, 300, 60, 300, 300, 300, 300, 300, 60, 300...

Note, the above examples are merely exemplary; other possible ratios for the current implementaion go up to 1 to 9.

The six main operations performed by the image post-processor of the present invention are described in greater detail below and with reference to Figures 1 and 3.

1. Line Merger

The line merger reads 1 to 9 lines from the appropriate image (color or black and white), optionally averaging some lines together before sending the resulting lines to the matrixer. The line merger essentially interleaves the two or more renditions of the scanned image into a single interleaved data stream to be sent to the host computer. By interleaving the two or more renditions of the scanned image on a line-by-line basis, the scanner does not have to save all of any rendition, as all renditions are generated, interleaved and sent to the host computer real time -- that is, on a somewhat line-by-line basis.

The line merger reads 1 to 9 lines of data from either the color image or the black and white image and averages some of the lines together to reduce the number of computations that must be done by the matrixer. In addition, the line merger can provide a simple form of y resolution prescaling by averaging additional adjacent lines together. For example, the line merger can read 9 lines from the buffer, average every 3 adjacent lines to reduce the effective y resolution by a factor of 3, average the two remaining outer lines for kernel symmetry, and then send the resulting 2 lines to the matrixer. Therefore, it has reduced the number of lines that must be processed by the matrixer from 9 to 2.

Any integer number of adjacent lines from 1 to N (where N is the number of lines read from the buffer - up to 9) can be averaged together. However, after adjacent lines have been averaged, the resulting number of lines must be 1, 3, 5, 7, or 9 so that symmetrical lines can be further averaged before being passed on to the matrixer. This means that if no adjacent lines are averaged, we can use either 1, 3, 5, 7, or 9 line kernels. However, if 2 or 3 adjacent lines are averaged, we can only use 1 or 3 line kernels. If 4 through 9 adjacent lines are averaged, we can only use a 1 line kernel.

Lines are averaged by adding the corresponding pixels of each line and dividing by a scale factor of 1, 2, 4, or 8. Therefore, exact averaging is obtained only when averaging together 1, 2, 4, or 8 lines. The line merger will overflow if the number of lines added together divided by the scale factor is greater than 2. Therefore, if 3, 5, 6, 7 or 9 line averaging is desired, the nearest lower scale factor should be used and the matrixer coefficients should be reduced to compensate for the extra "gain" introduced in the line merger. The only exception to this rule is when we are reading 9 lines from the buffer, averaging every adjacent 3 lines, and then using a 3 line kernel. In this case, the line merger will end up reducing 9 line to 2. One of these two lines (the one corresponding to the center of the kernel) will be obtained by adding the center 3 lines from the buffer. The other line will be obtained by adding the outer 6 lines from the buffer. However, only one divisor can be selected and must be selected such that no line coming out of the line merger has been multiplied by more than 2. Therefore, in this example, a divisor of 4 must be selected. Since the gain of the line merger is now 3/4, the matrixer coefficients may need to be increased to compensate.

The buffer image to be used (color or monochrome), the number of lines read from that image, the number of adjacent lines averaged, and the scale factor can be specified independently for each view to be generated.

2. Matrixer

The matrixer "mixes" the three colors of each pixel to produce a single color or three new colors before sending the lines on to the tone mapper.

3. Tone Mapper

The tonemapper uses lookup tables and interpolation to adjust image contrast and intensity before sending the data to the formatter. The output pixels are further transformed by the tonemapper which passes them through a lookup table which contains a transfer curve called the tonemap. Tonemaps are used to perform gamma correction, adjust contrast and intensity, enhance shadows, etc.

4. Formater

The formatter optionally thresholds or dithers the data before packing it into the specified data width. This stage can also invert the data, if desired, before sending it to the compression engine.

5. Compression Engine

The compression engine optionally uses the packbits algorithm to compress the line of image data before sending it to the host.

II. MULTIPLE VIEWS FROM ONE SCAN WINDOW

The preferred multiple image scanner has the ability to obtain two images from a single scan. The scanner command language (SCL) implementation of multiple image separates the concept of the scan window with the particular view of that window. SCL is the command language used for scanners and is commonly known. It should be noted that any command language could be used. However, in the preferred embodiment, SCL is used to generate multiple views from one scan window. The scan window is the physical area of the page that will be scanned (defined by the x and y position and the extents). The view of that scan window is the data contained inside the physical area (defined by the resolution, data type, mirror, inverse, etc.). There are scanner settings that are window and view independent (i.e., is ADF attached, serial number, etc.). The advantage of separating the view and the window is that it makes the definition of multiple views extensible to multiple window with multiple views. No other scanner has multiple image abilities or an implementation of multiple views. It should be noted that although in the present implementation of multiple views, a single scan is used to obtain multiple views of the same document, it is irrelevant whether one or more scans are used to obtain multiple views of the same document. It does not matter how the data is obtained (from one or more scans), the important point is that the SCL implementation of more than one view allows different data to represent the same space.

In a preferred embodiment, the Hewlett-Packard ScanJet 5p is utilized to allow the acquisition of two views of data simultaneously. However, it should be noted that any color scanner from any manufacturer could be modified to implement the present invention. While the region of the scan is the same, the data types and resolutions of the two views can vary. The limits on the variables set on the two views are described with the SetViewType macro in SCL as follows:

```
INT16 DualScan(INT16 Phase,
               PUINT8 pBufferView0,
               INT32 LengthView0,
               PINT32 pReceivedView0,
               PUINT8 pBufferView1,
               INT32 LengthView1,
               PINT32 pReceivedView1);
```

The parameters are set as indicated below:

Phase --Flag indicating if this is the first transfer in a sequence. Must be one of the following values:

SL_FIRST or SL_SCAN, the first buffer in the transfer,

SL_ADF_SCAN, the first buffer in an ADF transfer (Currently only supported by

HP1750A & HP5110A),

SL_NEXT, each additional buffer transfer;

PbufferView0 --Pointer to view 0 data buffer to receive the data;

LengthView0 --Size of the view 0 data buffer in bytes;

pReceivedView0 --Pointer to the actual number of bytes received from the scanner for view 0;

pBufferView1 --Pointer to view 1 data buffer to receive the data;

LengthView1 --Size of the view 1 buffer in bytes;

pReceivedView1 --Pointer to the actual number of bytes received from the scanner for view 1.

This function is nearly identical to the Scan() function, which is well known to those familiar with SCL, except that it supports the dual view mode of the scanner.

Some of the default settings for the two views of a scan are shown below in Table 1.

Parameter	Default: Image One	Default: Image Two
Data Type	0 (b/w threshold)	5 (color)
Data width	1 bit/pixel	24 bits/pixel
B/W Dither Pattern	0 (coarse fatting)	0(coarse fatting)
Coefficient Matrix	2 (green only)	0 (NTSC color)
Tone Map	0 (contrast/intensity)	0
Intensity	0	0
Contrast	0	0
Mirror	0 (off)	*
Auto Background	0 (off)	*
Inverse Image	0 (off)	0 (off)

Parameter	Default: Image One	Default: Image Two
X Resolution	300 ppi	100 ppi
Y Resolution	300 ppi	100 ppi
X Scale Factor	100%	100%
Y Scale Factor	100%	100%
X Location	0	*
Y Location	0	*
X Extent	2550 (pixels)	*
Y Extent	3300 (pixels)	*
Lamp	0 (off)	*
Scan Bar Position	stops where it is	*
Download Type	0 (b/w dither pattern)	0 (b/w dither pattern)
Downloaded 8x8 B/W Dither Pattern	none (old pattern is erased)	*
Downloaded 8 bit Tone Map	none (old map is erased)	*
Downloaded Color Matrix	none (old matrix is erased)	*
Downloaded Calibration Strip Parameter	none(old parameters erased)	*
Downloaded 16x16 B/W Dither Pattern	none (old pattern is erased)	*
Downloaded B/W Matrix	none (old matrix is erased)	*

Parameter	Default: Image One	Default: Image Two
Downloaded 10 bit Color Matrix	none(old parameters erased)	*
Downloaded RGB tone maps	none(old parameters erased)	*
Downloaded B/W tone maps	none(old parameters erased)	*
Downloaded RGB gains	none(old parameters erased)	*
Calibration Y-Start	-1 (white strip)	*
Calibration Strip parameters	0 (auto-select)	*
Calibration Mode	0(auto–calibration dependent on data width)	*
Speed Mode	0 (auto)	*
Compression	0 (off)	0 (off)
Select number views	1	NA
Selected view	1 (off)	NA
Calibration Strip Lower	0 (calculated)	*
Max. # Lines per Buffer	0 (use calculated)	*

In the above table "*" indicates that the parameter is not set independently for the two views. However, this could be modified to allow some or all of these parameters to be independently set for the two views. Also, these parameter settings and SCL commands reflect the implementation used by the inventors and is not intended to be the only possible combination of parameter settings or SCL commands that could enable the multiple views from a single scan concept of the present invention.

The following commands have been added to select the number of views to be sent by the scanner to the host computer:

Command:	Select Number of Views
Escape Sequence:	Esc*f#A
Inquire ID:	10466
Range:	1 - 2
Default:	1
Macro:	SetNumberofViews(x)
Send Command:	SendCommand(SL_NUM_OF_VIEWS,x)

Where x can be either 1 or 2.

This command will select how many views of the window will be sent by the scanner. The default will be one. The present implementation supported values are 1 = one view, 2 = two views. When two views are enabled, the data for the two views will be sent out interleaved on a line basis dependent on the ration of their Y resolutions. For example, if view one's Y resolution was 300 and view two's Y resolution was 150, then the data would alternate with two lines of view one's image and then one line of view two's image (except at the start; see examples above in the image post-processor section).

The select view command is used to set scanner settings for the view or views you want to work with as follows:

Command:	Select view
Escape Sequence:	Esc*f#B
Inquire ID:	10467
Range:	0 - 1
Default:	0
Macro:	SetViewType(x)
SendCommand:	SendCommand(SL_VIEW_TYPE, x)

Where x can be: 0 - for view one, or 1 - for view two.

The view must be selected prior to choosing the scanner settings. This command selects which view the subsequent SCL commands or inquires refer to. Supported values are: 0(view one) and 1(view two). The default is view one.

III. METHOD FOR PARSING MULTIPLE IMAGE SCAN DATA

Host computer software parses the interleaved multiple image data sent by the scanner. In the present implementation, the software program DualScan and its supporting modules permit a user unfamiliar with the details of the multiple image data format to easily parse multiple image data from the scanner into familiar tiff files, which can then be used with programs such as Photo Shop. DualScan is a software program sold by Hewlett-Packard Co. It is possible that one or both of the images will contain compressed data further complicating the parsing issue. The DualScan software allows use of a scanner with multiple image capability in its hardware by parsing the interleaved data stream received from the scanner. One skilled in the art could readily expand DualScan to include the parsing of multiple views of data, rather than the two views as described in the present implementation.

Figure 4 is a block diagram of a dual image data stream 136 being sent from the scanner (not shown) and parsed by a data stream parser 122 on the host computer (not shown) into view 1 data (monochrome image) 126 and view 2 data (color image) 124. The operation of the data stream parser will be further explained with reference to Figure 5, which is a flow chart of the data stream parsing function.

At the initial entry point, the resolution, compression state, and bytes per line for each of the images is imquired by the data parser at 200. Next image ratios for each view are calculated by the data parser at 202. Then interleaved image data is received by the data parser from the scanner at 204. The data parser then copies one line of scan data to the appropriate output buffer, decompressing the data if necessary at 206. The data parser then switches views (210) based on the image ratios calculated at 202. If no errors are detected and the parsing of the data stream is not finished (212), the data parser returns to 204 and receives more interleaved data from the scanner to be parsed. If an error is detected or the parsing of the data stream is completed at 212, the data parser returns this status to the calling function which takes an appropriate action. Error conditions may include: (1) output buffers full, (2) bad response from the scanner, (3) decompression engine out of data and scanner has no more data, or (4) scan completed. The steps in Figure 5 are also shown in source code in Appendix A.

It should be noted that although the implementation described herein describes a dual scan, dual views, and the parsing of two interleaved data streams, these concepts are readily extensible to multiple scan, multiple views, and the parsing of multiple interleaved data streams.

Also, although the implementation described herein illustrates the example of generating one high resolution gray scale image and one low resolution, high bit depth color image, which would work well for scanning documents containing text and color pictures, such as magazines, other scenarios are also possible and anticipated by the present invention. For example, generating at least one high resolution, low bit depth gray scale or monochromatic image and at least one low resolution, high bit depth gray scale or monochromatic image, which would work well for scanning documents such as monochromatic documents, black and white photographs, etc.

One could also generate multiple color scans one of high resolution, low bit depth and one of low resolution, high bit depth. Other combinatins of multiple image scans that could be generated are possible without deviating from the teachings and concepts of the present invention.

The foregoing description of the present invention has been presented for purposes of illustration and description. It is not intended to be exhaustive or to limit the invention to the precise form disclosed, and other modifications and variations may be possible in light of the above teachings. For example, the data stream parser is capable of parsing any interleaved data stream containing more than one set of data. The data stream does not have to be an scanned image data stream. Also, the SCL implementation of multiple views of the same window does not matter how the multiple views were obtained; the implementation is the same whether the views were obtained from one or more scans of the same area. The embodiment was chosen and described in order to best explain the principles of the invention and its practical application to thereby enable others skilled in the art to best utilize the invention in various embodiments and various modifications as are suited to the particular use contemplated. It is intended that the appended claims be construed to include other alternative embodiments of the invention except insofar as limited by the prior art.

United States Patent Application No. 08/959,604, from which this application claims priority, and the abstract accompanying this application are incorporated herein by reference.

CLAIMS

1. A method for generating multiple renditions of an image with a scanner, said method comprising the following steps:

(a) performing a single scan of said image with said scanner; and

(b) said scanner generating more than one rendition of said image obtained from said single scan of said image, said more than one rendition comprising at least one high resolution, low bit depth gray scale image and at least one low resolution, high bit depth gray scale image.

2. A method for generating multiple renditions of an image with a scanner, said method comprising the following steps:

(a) performing a single scan of said image with said scanner; and

(b) said scanner generating more than one rendition of said image obtained from said single scan of said image, said more than one rendition comprising at least one high resolution, low bit depth monochrome image and at least one low resolution, high bit depth monochrome image.

3. A method for generating multiple renditions of an image with a scanner, said method comprising the following steps:

(a) performing a single scan of said image with said scanner; and

(b) said scanner generating more than one rendition of said image obtained from said single scan of said image, said more than one rendition comprising at least one high resolution, low bit depth color image and at least one low resolution, high bit depth color image.

The Patent Office

Application No:	GB 9811620.5	**Examiner:**	John Donaldson
Claims searched:	1 to 3	**Date of search:**	12 October 1998

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.P): H4F(FCA, FCCB, FCCX, FCCY, FFF, FFS, FFX); G4R(RET, REX)

Int Cl (Ed.6): G06K 9/00, 9/20, 9/60, 9/78; H04N 1/00, 1/04, 1/40, 1/41, 1/46, 1/48, 1/64

Other: Online:WPI

Documents considered to be relevant:

Category	Identity of document and relevant passage		Relevant to claims
A, E	GB 2321361 A	(HEWLETT-PACKARD), see abstract	-
A	GB 2143702 A	(DAINIPPON), see abstract, page 3, lines 56 to 62	-

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art.
Y	Document indicating lack of inventive step if combined with one or more other documents of same category.	P	Document published on or after the declared priority date but before the filing date of this invention.
&	Member of the same patent family	E	Patent document published on or after, but with priority date earlier than, the filing date of this application.

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US98/17580

A. CLASSIFICATION OF SUBJECT MATTER

IPC(6) :GO6K 9/36, 9/46
US CL :382/248, 240, 166
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 382/248, 240, 162, 166, 249, 232, 233; 348/398, 403, 420, 438

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

APS, IEEE DATABASE
search terms: fractal, wavelet

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	RINALDO et al. An Improved Wavelet-Fractal Coder. Circuits and Systems, 1994 IEEE International Symposium. 1994. Vol. 3. pages 113-116.	1-34
A	MALLAT et al. Characterization of Signals from Multiscale Edges. IEEE Transactions on Pattern Analysis and Machine Intelligence. July 1992. Vol. 14. No. 7. pages 710 - 732.	1-34

□ Further documents are listed in the continuation of Box C. □ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search 23 NOVEMBER 1998	Date of mailing of the international search report 31 DEC 1998
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer BHAVESH MEHTA Joni Hill Telephone No. (703) 308-5246